A NUMERICAL METHOD FOR TRACKING A MOVING WATER-TABLE

E.K. Clearwater, M.J. O'Sullivan and R.A. Archer

Department of Engineering Science, University of Auckland, Private Bag 92019, Auckland 1142, New Zealand

emily.clearwater@auckand.ac.nz

Keywords: *Water table, TOUGH2, moving interface, numerical algorithm*

ABSTRACT

State-of-the-art software used for numerical modelling of non-isothermal, two-phase flow in geothermal systems does not allow for the accurate tracking of the water table over time. Current approaches use either a fixed water table or an air/water model where the location of the water table is inferred from the air mass fraction in shallow blocks – an approach which often suffers from convergence issues and whose accuracy is dependent on grid resolution.

The present paper first describes the current approaches used in geothermal and groundwater contexts, and then discusses two new methods for tracking the movement of the water table, based around the TOUGH2 code. The methods use a fully saturated, water-only, model where the top surface of the grid moves at each time step as required from a mass balance calculation (derived either through iteration or a flux approximation).

Further development and extension of the two methods, plus other mass balance approximation methods, will be explored in order to extend water table tracking to two-phase, non-isothermal flow.

1. INTRODUCTION

TOUGH2 (Pruess et al. 1999) is the worldwide standard code used to model the movement of mass and heat in geothermal reservoirs. Different equations of state (EOS) are used to describe the dependence of fluid properties on temperature and pressure, as well as the interaction between the components moving throughout the system (e.g. air, water, CO_2).

Most geothermal reservoir models, in order to avoid the complexity of non-isothermal air/water interaction, choose to ignore the vadose zone, and set the water table as a fixed top surface of the model, where ambient temperature and pressure conditions are assumed, with a "wet" atmosphere so water can move in and out of the model. Some recent examples of this type of model are (Romagnoli *et al.* 2010) and (Feather and Malate 2013). For some systems this can be an appropriate assumption as the water table may be shallow enough so that air movement and heat transfer through the vadose zone can be ignored. The temperature of the water table may indeed be close to the ambient temperature and the position of the water table may not be changing significantly over time. The setting of the top of the model at the water table allows model complexity and the simulation time to be greatly reduced. However, there are several situations where this approximation is not satisfactory, e.g.:

(i) Modelling areas where the water table is deep and the temperature may be above ambient,
(ii) Capturing the behaviour of surface features,
(iii) Modelling environmental effects such as subsidence.

Thus to understand situations where deep production influences the shallow water table movement, a more sophisticated model is required, particularly one that explicitly includes the vadose zone and can deal with a moving water table.

Several models developed at the University of Auckland have included the movement of the water table, for example: Wairakei (Mannington *et al.* 2000), Ohaaki (Clearwater *et al.* 2014) and Rotorua (Ratouis *et al.* 2014). For Wairakei an air/water equation of state (EOS) was used whereas for Ohaaki a CO_2/water EOS was used and for Rotorua both an air/water and a CO_2/NaCl/water EOS were used.

For these models the top of the model is located at the topographical land surface and ambient atmospheric conditions are assumed (with a "dry" atmosphere containing air and water vapour). Thus air and water vapour can move into the model, and water or air and water vapour can flow out. For this type of model the unsaturated zone is included, and the water table can move freely. However, thin layers must be used in the shallow part of the model in order to accurately resolve the location of the water table.

By including the vadose zone these air/water models represent the shallow zone much more accurately than the fixed water table models and hence are able to provide better understanding of changes in surface flow. However, their accuracy is dependent on grid resolution. Grid refinement leads to increased model complexity and a very large computational run-time, and the gas/water models often experience convergence and performance issues (O'Sullivan *et al.* 2013).

Convergence issues with air/water or CO2/water models have recently been investigated and the run-times improved (O'Sullivan *et al.* 2014), (O'Sullivan *et al.* 2013). However, an accurate numerical method that can accurately track the movement of the water table in a geothermal reservoir model is still required.

2. TRACKING MOVING INTERFACES

Many scientific and engineering problems involve a moving interface between two fluids of different densities (including phase-change fronts). The methods used to resolve these free surface flows are complicated as they address complex phenomena such as fluids folding over themselves, droplets forming or explosions. Because flow through a geothermal reservoir is slow and the water table is likely to be relatively smooth in shape, the problem of tracking the movement of the water table above a geothermal reservoir should be more straightforward. One can look towards current developments in groundwater modelling as a starting point and later for dealing with non-isothermal flow and boiling the more complicated methods used in aeronautics, implosion, underwater explosions and other problems may be appropriate.

The main issue in moving from groundwater problems to a geothermal problem is that the groundwater flow is

isothermal and the single-phase flow equations are directly solved in terms of pressure head and water table elevation. Whereas a TOUGH2 simulation of a geothermal problem solves for pressure, temperature (or vapour saturation) and mass and energy fluxes, and may involve two-phase flow. The water table can then be located from a transition from two-phase conditions to all-liquid conditions. The mass flows near this point can be used to deduce the movement of the water table.

The intention of the research described in this paper is to improve the ability of TOUGH2 to track the water table. Therefore, a change to another type of discretisation or to a meshless method is not envisaged, instead the focus lies in developing new ideas easily incorporated into the current TOUGH2 solution structure

2.1 Eularian Frameworks

TOUGH2 is based on a Eularian framework where mass and energy difference approximations are applied to a fixed grid. In the case of the finite volume method used in TOUGH2, the primary variables (pressure and temperature) are calculated at block centres while fluxes are calculated at block boundaries.

As mentioned in Section 1, the simplest way to track a phase change interface within a fixed grid structure is to use mesh refinement, but this approach can become very computationally costly, especially for tracking an interface that moves large distances, as many layers of the grid must then be refined. At best, utilising this technique only results in a layer-by-layer representation of the movement of the water table, rather than continuous, accurate tracking.

(Keating and Zyvoloski 2009) modified the numerical simulator FEHM (Zyvoloski 2007) to allow continuous tracking of the water table within a Eularian framework. FEHM is a multi-phase flow simulator using a conservation of mass and energy to solve for the pressure of each component (e.g. water and air). Keating and Zyvoloski's technique uses a simplified air/water method which avoids discontinuities in the saturation/pressure relationship (i.e. convergence problems associated with grid blocks switching across the phase-change boundary), while still giving a saturation dependent result. This is achieved by solving a multi-phase mass conservation equation, but assuming constant air pressure.

For a block containing the water table they introduce a pseudo liquid saturation which measures how "full" the block is of water. The mass balance equation is solved for water pressure, p_w, the sole primary variable. Then the water pressure is used to derive pressure head and then the pseudo saturation is calculated as follows. First the head is calculated assuming an approximate hydrostatic pressure profile in the block:

$$h = \frac{p_w}{\rho g} + z$$

Here h is pressure head (which determines the water table level), ρ density and g gravity. Pressure head and the elevation ranges within a grid block, z_2 and z_1, are then used to calculate saturation:

$$S_l = \frac{h - z_2}{z_1 - z_2}$$

Here S_l is the pseudo liquid saturation, which is then entered back into the mass balance equation, both in the accumulation term and in calculating the effective permeability for horizontal flow between partly full blocks.

The intention of Keating and Zyvoloski was to create a method of solving for the water table that was more numerically stable than usual groundwater simulators (i.e. avoiding "dry cells" such as those used in MODFLOW 2000 (Harbaugh *et al*. 2000) and utilising a NR iterative scheme rather than Picard iteration). Comparisons of their approach with the equivalent MODFLOW simulations show improved accuracy of the water table location and better convergence properties. Keating and Zyvoloski also compare their method with FEHM equivalent air/water two-phase methods and show that their water table tracking method more accurately represents the pressure near the water table, as well as being more efficient and stable. A drawback to their method is that a problem-dependent parameter needs to be utilised in order to scale permeability in partially saturated cells, although their examples show the same value for this parameter works for most problems.

2.2 Lagrangian and Mixed Frameworks

Lagrangian and mixed approaches allow meshes (either the mesh the governing equations are solved on or a reference mesh) to deform and move as the interface moves. These methods have advantages of being easy to implement as the whole domain does not need to be re-meshed at each time step, but can be difficult to implement in non-homogenous domains, and may create undesirable aspect ratios in some elements.

(Knupp 1996) utilized a moving coordinate system for the solution of free-surface (fully saturated only) groundwater flow problems. The finite volume method was utilised to discretize the free-surface flow equations, with backwards time differencing. In this case, pressure head is the primary unknown and so once the solution at each time step converges, the new water table elevation is easily calculated. After each time step, Knupp applies grid movement within each column (no horizontal movement of the nodes are made), and rock properties are vertically interpolated from the original stratigraphic domain on to the new block centres. Although the method attempts to have the grid fit the stratigraphy, and only nodes near the water table are moved, the modified model stratigraphy does not exactly match the original, especially when blocks are deformed over large vertical distances and the stratigraphy is very heterogeneous. Knupp's moving mesh algorithm is implemented in the groundwater modelling code SECO-3D (Roache 1993)..

(Crowe *et al*. 1999) presented a method which again solves the free-surface flow equations for groundwater flow whilst tracking the water table. However, their method is aimed at more accurately approximating heterogeneous and complex stratigraphy. Crowe *et al.*, solve a standard finite element discretization (nodes situated at boundaries) of the free-surface groundwater equations, giving directly a solution for pressure head and water table elevation. At each time step, nodes that lie on the water table are moved to a new elevation, creating a new vertical spacing in the top layer of elements. If the new vertical spacing is more than ¼ that of regular grid spacing, Δz, a new layer is added with the original stratigraphy at the corresponding elevation applied. This layer adding approach gives a maximum layer extension of $5/4\,\Delta z$. When the maximum layer extension

occurs, an error is introduced if there is a difference in rock properties between the previous and current water table surface layers.

Once nodes are moved, numerical convergence is tested for by calculating a residual between the head at the node and the elevation of the node. Once this residual is within a defined tolerance the simulation moves on to the next time step. One problem encountered by Crowe *et al.*, with their method was a convergence issue when the water table moved through units that had a large contrast in rock properties. Although they offered a numerical solution to this issue, they found the best way to obtain a convergent solution for all problems was manual intervention and manipulation of rock properties to reduce the contrast.

3. NUMERICAL METHOD

The ultimate goal is to have a non-isothermal, multiphase, saturated/unsaturated model which tracks the water table as part of the TOUGH2 solution process. In order to achieve this goal, simple models and numerical algorithms are being explored first, and are built upon as they are tested and validated. The methods described here assume one dimensional, isothermal conditions and thus density and other water properties are not changing with time, energy conservation does not need to be considered, and there are no horizontal flows. The water table surface is set at the top of the model, and the methods used do not take into account the vadose zone or capillary effects. The methods are focused on sub-surface fluid movement so surface flows and run-off are not considered and it is assumed that the infiltration rate is known. The water table is assumed to correspond to a sharp discontinuity in fluid properties. Hence, the governing equations described in this work are those for a liquid saturated, isothermal model.

The new numerical methods described here employ a very similar algorithmic procedure to Crowe *et al.*, and use a layer building technique to update the grid, but because of the difference in governing equations and discretization method, they incorporate different numerical strategies.

3.1 Governing Equations

TOUGH2 solves mass and energy balance equations using an integrated finite difference technique. For this method, the region of interest is divided into blocks or elements where the ith block has a volume V_i and connection area a_{ij} to the adjacent jth block. As time is incremented, the mass and energy flux in and out of each element is calculated. The difference equations are fully implicit as mass and energy fluxes are evaluated at the new time step. Mobilities and enthalpies are upstream weighted, and interface permeabilities and conductivities are harmonically weighted. Because the problems considered in this project so far are isothermal, the governing equations given below are for isothermal all-liquid conditions. The discretized mass balance (grid shown in Figure 1) can be written as:

$$V_i\left(A_{mi}^{n+1} - A_{mi}^{n}\right) = \sum_j a_{ij} F_{mij}^{n+1} \Delta t_n + Q_{mi}^{n+1} \Delta t_n \qquad (1)$$

Here A_{mi} denotes the mass of fluid per unit volume of reservoir in block i and Δt_n the duration of the nth time step. A_{mi} is defined for isothermal liquid conditions by:

$$A_m = \varphi\rho \qquad (2)$$

Here φ is porosity and ρ density. The mass flux F_{mij}^{n+1}, from block i to block j evaluated at the end of the $(n+1)$th time step, is given by the discrete version of Darcy's Law:

$$F_{mij}^{n+1} = \left(\frac{k}{v}\right)_{ij}^{n+1} \left[\frac{P_j^{n+1} - P_i^{n+1}}{d_{ij}} - \rho_{ij}^{n+1} g_{ij}\right] \qquad (3)$$

Here k is permeability, v is viscosity, P pressure and g gravity. The distance between block centres, d_{ij}, is the sum of distances d_i and d_j from the centres of the ith and jth block to their connecting interface, respectively.

Similarly, Q_{mi}^{n+1} is the mass production or injection from or into block i evaluated at the end of the $(n+1)$th time step.

Both approaches to tracking the water table (described in Sections 3.2 and 3.2 below) are based on the calculation of the mass flow into or out of the top surface of the model (the water table surface). The correct position of the water table is one where fluid is no longer moving into or out of the top surface of the water table block. For a given time step this position is found when the sum of fluxes across the top surface, F_{m0i}^{n+1}, is zero. The top surface of the grid (the water table elevation for the previous time step) is then moved to lie at the correct water table position for the current time step.

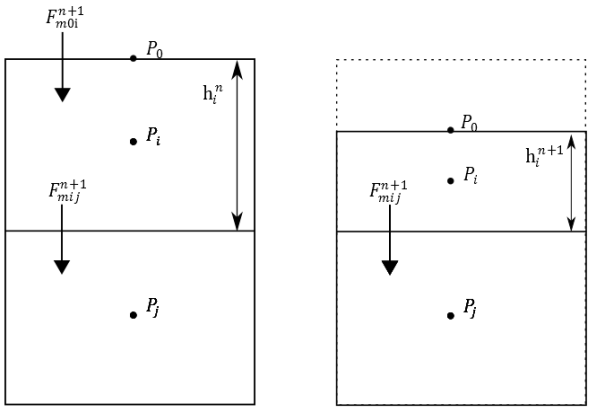

Figure 1: Left, TOUGH2 mass balance on block i at time t=n. Right, mass balance on moving water table block i at time t=n+1.

3.2 Method 1 - Iterative Method

For this method, the top surface of the grid (the water table elevation) is iteratively moved and the time step re-run until the correct position is found for the current time step. The correct position moves the water table surface so a new volume is created in which there is no flux across the top surface of the water table block. The following derivation describes how one can calculate the correct volume required for a time step without interfering with the TOUGH2 code.

Using the left image in Figure 1 as an example, a standard TOUGH2 mass balance on water table block i at time step Δt can be written as (from equation (1):

$$V_i^{n+1,k} A_{mi}^{n+1,k} - V_i^{n} A_{mi}^{n} = a_{0i} F_{m0i}^{n+1,k} \Delta t_n - a_{ij} F_{mij}^{n+1,k} \Delta t_n + Q_{mi}^{n+1,k} \Delta t_n \qquad (4)$$

For this method, we iteratively adjust $V_i^{n+1,k}$ and re-solve equation (4) until $F_{m0i}^{n+1,k} = 0$ (i.e. the image on the right in

Figure 1). Because the new volume is fed back into TOUGH2 to be re-solved for iteration k+1 of time step Δt, TOUGH2 is actually solving the following:

$$V_i^{n+1,k}A_{mi}^{n+1,k} - V_i^{n+1,k}A_{mi}^{n} = a_{oi}F_{m0i}^{n+1,k}\Delta t_n - a_{ij}F_{mij}^{n+1,k}\Delta t_n + Q_{mi}^{n+1,k}\Delta t_n \quad (5)$$

The iterative process starts with a standard solve of the TOUGH2 time step, with

$$V_i^{n+1,0} = V_i^{n} \quad (6)$$

Thus, equation (5) becomes:

$$V_i^{n}A_{mi}^{n+1,1} - V_i^{n}A_{mi}^{n} = a_{oi}F_{m0i}^{n+1,1}\Delta t_n - a_{ij}F_{mij}^{n+1,1}\Delta t_n + Q_{mi}^{n+1,1}\Delta t_n \quad (7)$$

As described earlier, we want to solve for $F_{m0i}^{n+1,k} = 0$. Hence, we really wish to solve:

$$V_i^{n+1,1}A_{mi}^{n+1,1} - V_i^{n}A_{mi}^{n} = 0 - a_{ij}F_{mij}^{n+1,1}\Delta t_n + Q_{mi}^{n+1,1}\Delta t_n \quad (8)$$

In which

$$V_i^{n+1,1} = V_i^{n} + a_{0i}\Delta h^{(1)} \quad (9)$$

And

$$\Delta h^{(1)} = h_i^{n+1,1} - h_i^{n} \quad (10)$$

As shown in Figure 1.

Substituting equation (9) into equation (8) results in:

$$V_i^{n}A_{mi}^{n+1,1} - V_i^{n}A_{mi}^{n} = -A_{mi}^{n+1,1}a_{0i}\Delta h^{(1)} - a_{ij}F_{mij}^{n+1,1}\Delta t_n + Q_{mi}^{n+1,1}\Delta t_n \quad (11)$$

By comparing equations (7) and (11), we find an equation describing $\Delta h^{(1)}$:

$$\Delta h^{(1)} = -\frac{F_{m0i}^{n+1,1}}{A_{mi}^{n+1,1}}\Delta t_n \quad (12)$$

To proceed past one iteration, and iterate towards a volume $V_i^{n+1,k}$ which results in $F_{m0i}^{n+1,1}$=0, we resolve equation (5) using TOUGH2. The volume $V_i^{n+1,k}$ is now the new updated volume found from equation (12). Again (similarly as for equation (8)), what we really want to solve rather than equation (5), is:

$$V_i^{n+1,k}A_{mi}^{n+1,k} - V_i^{n}A_{mi}^{n} = 0 - a_{ij}F_{mij}^{n+1,k}\Delta t_n + Q_{mi}^{n+1,k}\Delta t_n \quad (13)$$

With

$$V_i^{n+1,k} = V_i^{n} + a_{0i}\Delta h^{(k)} \quad (14)$$

Substituting equation (14) into (5) we get:

$$V_i^{n+1,k}A_{mi}^{n+1,k} - V_i^{n}A_{mi}^{n} = a_{0i}\Delta h^{(k)}A_{mi}^{n} + a_{0i}F_{m0i}^{n+1,k}\Delta t_n - a_{ij}F_{mij}^{n+1,k}\Delta t_n + Q_{mi}^{n+1,k}\Delta t_n \quad (15)$$

Comparing equation (15) and equation (13), the first two terms on the RHS of equation (15) should balance out. When they do, the volume is correct, the water table surface is in the correct position, and the next time step can be solved. If the terms do not balance out, $\Delta h^{(k+1)}$ may be calculated using:

$$\Delta h^{(k+1)} = -\frac{a_{0i}F_{m0i}^{n+1,k}\Delta t_n}{a_{0i}A_{mi}^{n+1,k}} \quad (16)$$

And

$$\Delta h^{(k+1)} = h_i^{n+1,k+1} - h_i^{n} \quad (17)$$

The procedure is then iterated by applying $h_i^{n+1,k+1}$ to block volumes and connection distances, and re-solving the time step.

The steps of the method are as follows:

1. New TOUGH2 time step. The top surface of the model is set at the water table surface, h_i^n, $k = 0$.

2. Solve one TOUGH2 time step, Δt_n , $k = k + 1$.

3. Extract the mass flow through the water table surface, $F_{m0i}^{n+1,k}$, from the TOUGH2 results.

4. Evaluate $\Delta h^{(1)}$ using Equation (12)

5. Use the value for $\Delta h^{(1)}$ and equation (10) to move the water table surface to $h_i^{n+1,k+1}$. Update the block volumes, connection distances and block centre locations in the TOUGH2 file. Block properties for this updated model are taken from the initial input data file and grid, so that the original stratigraphy and parameters are preserved.

6. Repeat Step 2 and Step 3 using the updated volume.

7. Evaluate:

   7.1 $a_{0i}\Delta h^{(k)}A_{mi}^{n} - a_{0i}F_{m0i}^{n+1,k}\Delta t_n \leq tol$: stay with this solution, ready to move on to the next TOUGH2 time step. Go to Step 1.

   7.2 $a_{0i}\Delta h^{(k)}A_{mi}^{n} - a_{0i}F_{m0i}^{n+1,k}\Delta t_n \not\leq tol$: the elevation of the water table block is wrong and block volumes need to be adjusted. Continue to Step 8.

8. Use equations (16) and (17) to find $\Delta h^{(k+1)}$ and move the water table surface to $h_i^{n+1,k+1}$. Update the block volumes, connection distances and block centre locations in the TOUGH2 file. Go to Step 6.

Step 6 – Step 8 are repeated until the convergence criteria in Step 7.2 is met. A diagram of this workflow is shown in Figure 2. This method has been implemented in 1-D (see Section 4) with good results.

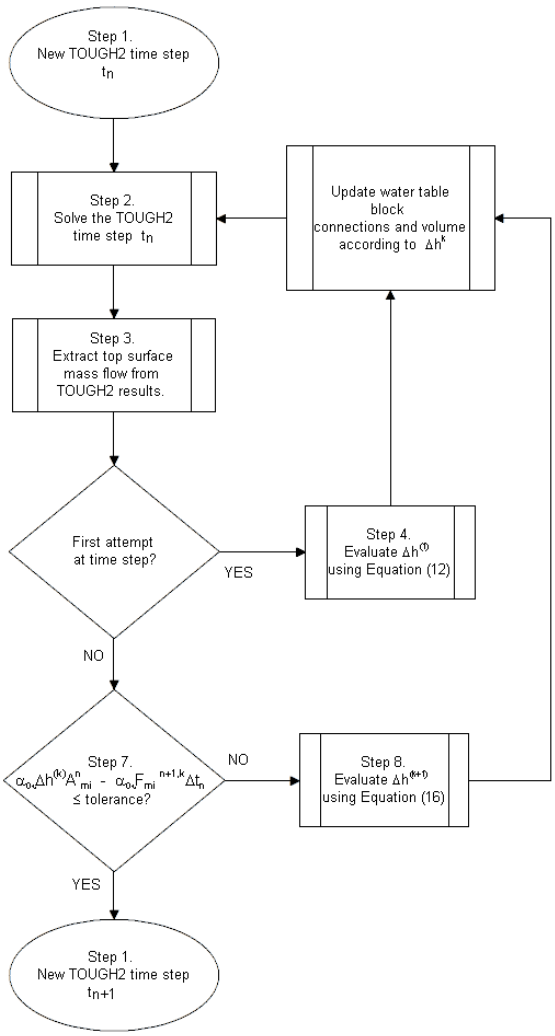

Figure 2: Algorithm for Method 1 – Iterative Method

3.2 Method 2 – Flux Approximation Method

This method essentially takes one step of the above iterative method. It assumes the flux out the top surface of the model at the first iteration, $F_{m0i}^{n+1,1}$, is related to the required change of height of the water table block by equation (12). In this case, equation (8) (the mass balance we want TOUGH2 to solve) becomes:

$$V_i^{n+1}A_{mi}^{n+1} - V_i^{n}A_{mi}^{n} = 0 - a_{ij}F_{mij}^{n+1}\Delta t_n + Q_{mi}^{n+1}\Delta t_n \quad (18)$$

Where

$$V_i^{n+1} = V_i^{n} - a_{0i}\Delta h \quad (19)$$

And

$$\Delta h = h_i^{n+1} - h_i^{n}$$

Substituting (19) into (18):

$$V_i^{n}(A_{mi}^{n+1} - A_{mi}^{n}) = -a_{0i}\Delta h A_{mi}^{n+1} - a_{ij}F_{mij}^{n+1}\Delta t_n + Q_{mi}^{n+1}\Delta t_n \quad (20)$$

If we then ignore the moving boundary, a TOUGH2 mass balance for the original volume V_i^n is (as for equation (7)):

$$V_i^{n}(A_{mi}^{n+1} - A_{mi}^{n}) = a_{0i}F_{m0i}^{n+1}\Delta t_n - a_{ij}F_{mij}^{n+1}\Delta t_n + Q_{mi}^{n+1}\Delta t_n \quad (21)$$

By comparing equation (21) to (20), it is clear that the Δh term can be approximated by the flux across the top surface:

$$a_{0i}\Delta h A_{mi}^{n+1} = -a_{ij}F_{m0i}^{n+1}\Delta t_n \quad (14)$$

The fluid storage calculation in this method assumes negligible change in density between time steps. Hence, the fluid properties at the previous time step are used:

$$A_{mi}^{n+1} \approx A_{mi}^{n}$$

This approximation allows equation (2) to be substituted into (14):

$$a_{oi}\Delta h \varphi_i^n \rho_i^n = \Delta t_{n+1} a_{ij}F_{m0i}^{n+1} \quad (15)$$

Rearranging (15) to make the change in block height the subject (as this is the variable required to update the TOUGH2 input file block parameters) results in:

$$\Delta h = \frac{\Delta t_n F_{m0i}^{n+1}}{\varphi_i^n \rho_i^n} \quad (16)$$

There are two assumptions made for this method. One, there is a negligible change of density between time steps, and two, the flux across the top surface can be used to find the change in height of the top block. These assumptions remove the need for an iterative approach as used in Method 1, meaning there is no convergence issue and it is computationally faster. If the time step Δt_n is chosen appropriately, each time step will capture small changes in mass flow (and hence changes in the top surface flux), and so errors arising from these assumptions will be small. For the isothermal cases considered here, this is a viable method. However, for two-phase flow these assumptions may produce large errors in the location of the water table.

4. EXAMPLE –GRAVITY DRAINAGE

To demonstrate the methods described in Section 3, a simple example of gravity drainage of water down a column of soil is considered. For comparison, the equivalent air/water EOS model in TOUGH2 is also used to solve the problem.

4.1 Model set-up

The numerical simulations were carried out using AUTOUGH2 ((Bullivant 1990) and (Yeh *et al.* 2012)) a version of TOUGH2 (Pruess *et al.* 1999) developed at the University of Auckland. The moving water table simulations use EOS1 (water only) and the air/water model uses EOS3. The model grid consists of one vertical column with a total height of 0.92m. There are two rock-types, coarse and fine sand. The fine sand overlays the coarse sand, and each fills half the column, i.e. a depth of 0.46m. Two grid resolutions are used: Grid 1 is discretised into 46 layers, each of which is 0.02m thick, whereas Grid 2 is coarser, with 23 layers, each of which is 0.04m thick. Because of the discretization used with Grid 2, the fine sand fills the column from the surface down to 0.48m.

The properties of the sands are given in Table 1, and the grid structure and rock-type assignment are shown in Figures 3 and 4, respectively.

Table 1: Formation properties for the drainage model

	fine sand	*coarse sand*
Rock grain density *(kg/m²)*	2.08E+03	2.08E+03
Porosity	3.00E-01	3.40E-01
Permeability (isotropic) *(m^2)*	7.27E-11	1.89E-10
Heat conductivity *(W/m K)*	3.00E+00	3.00E+00
Specific heat *(J/kg K)*	8.30E+02	8.30E+02

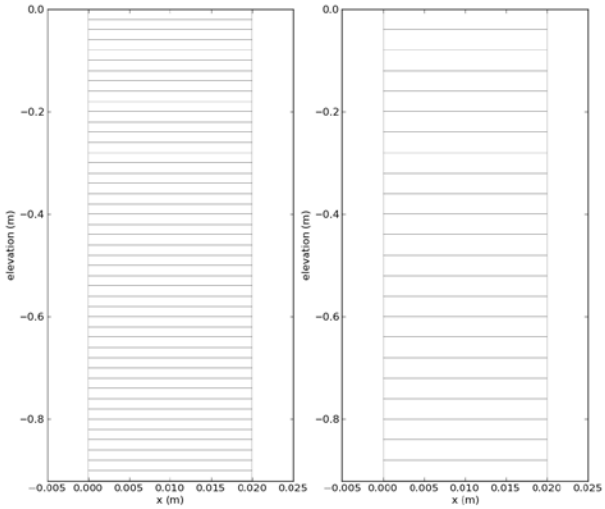

Figure 3: Left, Gravity drainage model Grid 1. Right, Gravity drainage model Grid 2.

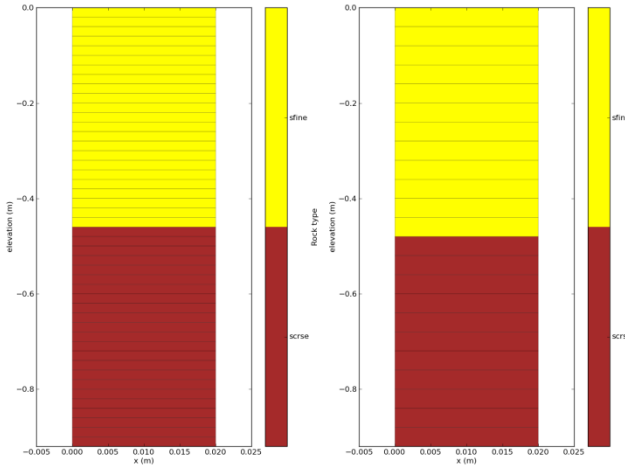

Figure 4: Rock-type allocation. Left, Gravity drainage model Grid 1. Right, Gravity drainage model Grid 2.

4.1.1 Boundary and initial conditions – moving water table grids

For the moving water table models, the side boundaries are closed. A constant atmospheric pressure boundary condition is applied to the top block in the computational model, ie at the water table. This is implemented by connecting a large volume block filled with water with a very small connection distance to the top grid block. The pressure of the large volume block is atmospheric pressure (101.325kPa). Initially the water table is set at the top of the column, although throughout the simulation the water table moves and the computational grid is changed. Thus, the atmosphere boundary block moves with the water table.

Initially, all blocks are fully saturated with water and pressure is at hydrostatic equilibrium. This was achieved by using a hand calculation for head at each block centre, using the density of water at 25°C.

4.1.2 Boundary and initial conditions – air/water grids

In the air/water model the side boundaries are also closed. The large atmosphere block at the top of the grid is filled with air and water vapour at atmospheric pressure. As for the moving water table models, initially all grid blocks are fully saturated with water and pressure is at hydrostatic equilibrium.

Because the moving water table methods assume a sharp transition of density across the water table, and the air water model is being used as a comparison, no capillary effects were included in the air/water model.

The relative permeability function (shown in Figure 5) is linear with coefficients chosen such that liquid was fully mobile and gas very immobile (in TOUGH2 terms, RP = 0, 0.999, 0.001, 0.9991). Keeping the gas immobile was necessary in order to create a sharp transition of saturation across the air/water interface. The choice of relative permeability function creates the relatively sharp transition in saturation seen in Figure 7.

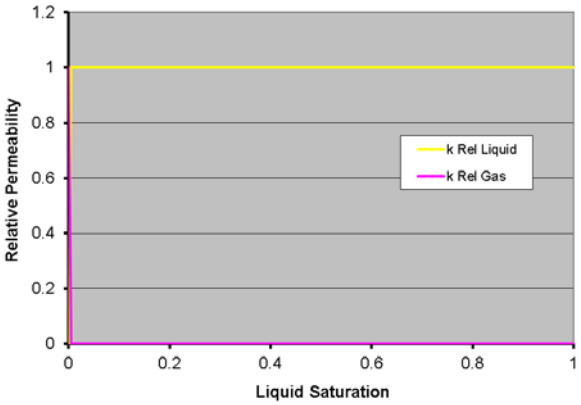

Figure 5: Relative permeability function for the air/water model.

4.1.3 Simulation – all models

Drainage begins when the simulation starts. At the beginning of the simulation, the pressure at the base of the model is set to atmospheric by attaching a large atmosphere block filled with water at 101.325kPa (similar to that attached to the top grid block) to the bottom grid block. This allows the fluid to begin draining out of the bottom block of the column.

The simulation was run for 150 seconds at a time step of no larger than 0.16 seconds. Over the 150 seconds simulated, the column water table level drained down to -0.73m in Grid 1, and -0.68m in Grid 2. For the air/water model, air from the atmosphere boundary block moves downwards into the column replacing water as it drains out the bottom. For the moving water table models, the atmosphere boundary block stays attached to the water table grid block as that block moves downwards.

4.2 Model results.

Results for elevation of the water table versus time using Grid 1 are shown in Figure 6, and the Grid 2 results are shown in Figures 8 and 9. The location of the water table in the air/water model is determined by first finding the bottom-most layer in which gas saturation is more than 0.001. Then a hydrostatic pressure profile is calculated upwards from the centre of this block and the level where atmospheric pressure is reached is set as the water table location.

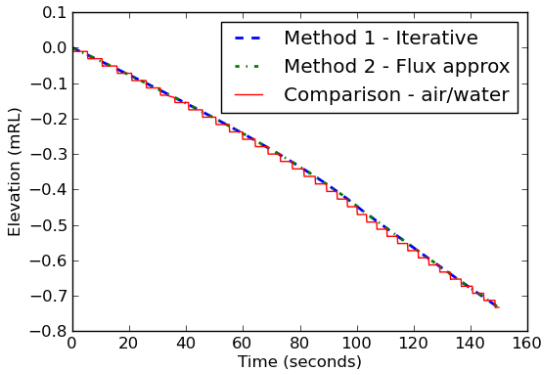

Figure 6: Gravity drainage model results – Grid 1.

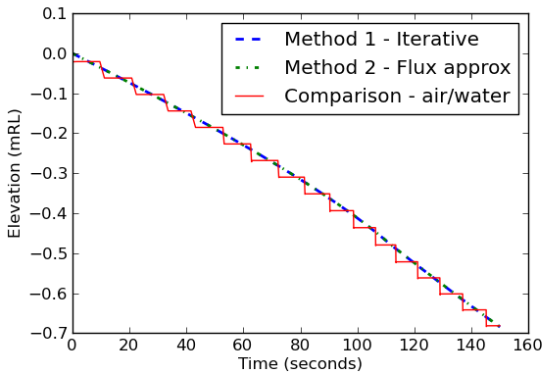

Figure 7: Gravity drainage model results, Grid 2.

For the Grid 1 the results from all three methods look very similar. The fine resolution of the grid enables the air/water method to give a good match to the other methods. The Flux Approximation (Method 2) solution is very close to that of the Iterative Method (Method 1) solution. At each time step, the Method 2 result differs to that of the final solve of Method 1 (which takes ~2 extra iterations to converge to a tolerance of 1.0e-15) by at most 6.0e-4. The air/water method follows the moving water table methods closely, and the result at 150 s has a difference of 1.0e-3 to the moving boundary solutions for Grid 1 and 3.0e-3 for Grid 2. However, due to the shape of the air/water solution, results are inaccurate at certain times during the simulation. The Grid 2 results (Figures 8 and 9) display this - the air/water method results are stepped and the size of the steps changes depending on the grid resolution.

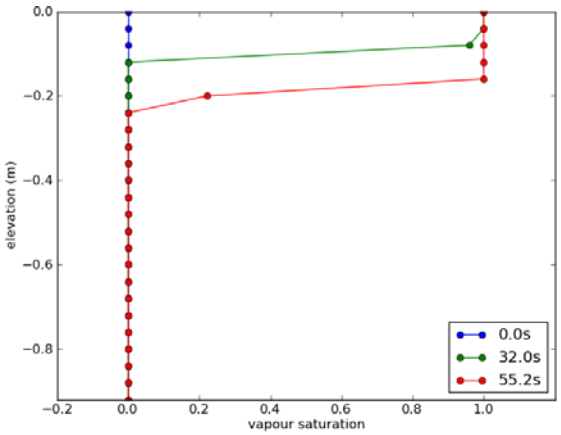

Figure 8: Vapour saturation vs depth for the air/water model using Grid 2 at times 0.0s, 32.0s and 55.2s.

The step-wise shape in the air- water model solution is due to saturation changes. Pressure in the water table block remains constant while air saturation increases - this is the flat part in each step of the air/water solution. The vertical part of each step of the air/water solution corresponds to when the vapour saturation in the current water table block increases beyond 0.001, and the water table is assumed to move down to the next layer. The value of 0.001 is arbitrary and was chosen by observing plots of the vapour saturation over time (as seen in Figure 7). The step-wise shape of the air/water solution shows how the air/water method is limited in terms of accuracy by the grid resolution. A detailed plot over the time period 85s – 110s is shown in Figure 9. The time at which the step has just dropped down, and the water table has moved to a new layer, are when the air/water model solution is least accurate. Then, as the pressure stays constant and the air saturation increases, the air/water solution nears the moving water table method solutions, until it drops again and the cycle repeats.

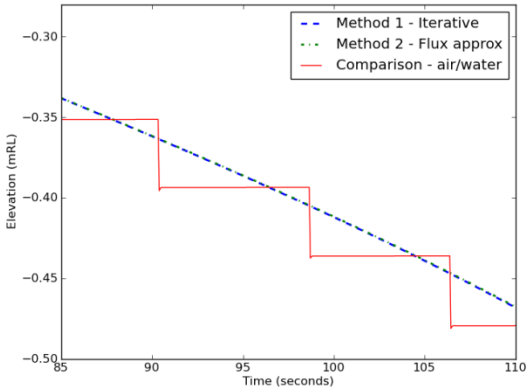

Figure 9: Gravity drainage model results detail, Grid 2.

Note also that the air/water method result fluctuates over time around the solutions from the other methods. Using the results shown in Figure 9 as an example, the point on the flat top of the step of the air/water solution, where the moving water table methods intersect the step, varies over the three steps shown in Figure 9. This may be a feature of the relative permeability used (which has no influence on the behavior of the other models as they are single phase).

Quantitative analysis and comparison of the computational speed between all three methods has not been undertaken, but the Flux Approximation Method is much quicker than

the Iterative Method (each time step is run once for Method 2, compared to an average of 3 times for Method 1). For this isothermal case, the air/water simulation speed is similar to the speed of the moving water table methods.

Larger, more complex models need to be investigated in order to ascertain which of the two moving boundary methods yields the best results. The accuracy of the Flux Approximation Method is not surprising for this simple, isothermal case, but it needs to be tested on more challenging non-isothermal problems. The limitations of both methods also need to be considered further. The Flux Approximation Method assumes constant fluid density, which cannot be assumed in a non-isothermal simulation, while the Iterative Method is very computationally costly, and it is not clear yet how to extend the method to 2D or 3D simulations including horizontal flow.

5. CONCLUSIONS AND FURTHER WORK

Numerical methods for tracking the water table in a TOUGH2 simulation were presented and results for simple isothermal problems show the methods to be working effectively. These methods are not limited in accuracy by the thickness of the model layers, as is currently the case in standard approaches to capturing the movement of a water table.

Work is continuing in understanding the computational error associated with the Flux Approximation Method, and reducing the computational expense associated with Iterative Method in 2-D and 3D problems. Further development and extension of both the Iterative and Flux Approximation methods, plus other solution methods, will be explored in order to extend water table tracking to two-phase, non-isothermal flow.

ACKNOWLEDGEMENTS

E.K Clearwater is funded by the Todd Foundation Energy Scholarship and a University of Auckland Doctoral Scholarship.

REFERENCES

Bullivant, D. P.: Making Mulkom/Tough Faster and Easier to Use. *Proc. TOUGH Workshop 1995*, Lawrence Berkeley Laboratory, Berkeley, California. (1990).

Clearwater, E. K., O'Sullivan, M. J., Newson, J. A., Brockbank, K. and Mannington, W.: An Update on the Numerical Model of the Ohaaki Geothermal System. *Proc. 36th New Zealand Geothermal Workshop.*, Universtiy of Auckland, Auckland, New Zealand. (2014).

Crowe, A. S., Shikaze, S. G. and Schwartz, F. W.: A Grid Generating Algorithm for Simulating a Fluctuating Water Table Boundary in Heterogeneous Unconfined Aquifers. *Advances in Water Resources*, **22** (6), 567 - 576, (1999)

Feather, B. M. and Malate, R. C. M.: Numerical Modelling of the Mita Geothermal Field. *Proc. 38th Workshop on Geothermal Reservoir Engineering*, Stanford University, Stanford, California. (2013).

Harbaugh, A. W., Banta, E. R., Hill, M. C. and MacDonald, M. G.: *Modflow 2000, the U.S. Geological Survey Modular Groundwater Model—User Guide to Modularization Concepts and the Ground-Water Flow Process.* USGS Open-File Report 90-392. (2000)

Keating, E. and Zyvoloski, G. A.: A Stable and Efficient Numerical Algorithm for Unconfined Aquifer Analysis. *Groundwater*, **47** (4), 569 - 579, (2009)

Knupp, P.: A Moving Mesh Algorithm for 3-D Regional Groundwater Flow with Water Table and Seepage Face. *Advances in Water Resources*, **19** (2), 83 - 96, (1996)

Mannington, W., O'Sullivan, M. J. and Bullivant, D. P.: An Air / Water Model of the Wairakei - Tauhara Geothermal System. *Proc. World Geothermal Congress*, Kyushu - Tohoku, Japan. (2000).

O'Sullivan, J. P., Croucher, A. E., Yeh, A. and O'Sullivan, M. J.: Improved Convergence for Air/Water and Co2-Water Tough2 Simulations. *Proc. 35th New Zealand Geothermal Workshop*, Ed. Rotorua, New Zealand. (2013).

O'Sullivan, J. P., Croucher, A. E., Yeh, A. and O'Sullivan, M. J.: Further Improvements in the Convergence of Tough2 Simulations. *Proc. 11th World Congress on Computational Mechanics (WCCM XI) joint with the 5th European Conference on Computational Mechanics (ECCM V) and 6th European Conference on Computational Fluid Dynamics (ECFD VI)*, Ed: Oñate, E., Oliver, J. and Huerta, A. Barcelona, Spain. (2014).

Pruess, K., Oldenburg, C. and Moridis, G.: *Tough2 User's Guide, Version 2.0*. Lawrence Berkeley National Laboratory Report LBNL-43134. (1999)

Ratouis, T. M. P., O'Sullivan, M. J. and O'Sullivan, J. P.: An Updated Numerical Model of Rotorua Geothermal Field. *Proc. Thirty-Ninth Workshop on Geothermal Reservoir Engineering*, Stanford University, Stanford, California. (2014).

Roache, P. J.: The Seco Suite of Codes for Site Performance Assessment. *High-Level Radioactive Waste Management: Proceedings of the fourth Annual International Conference*, Las Vegas, Nevada. (1993).

Romagnoli, P., Arias, A., Barelli, A., Cei, M. and Casini, M.: An Updated Numerical Model of the Larderello–Travale Geothermal System, Italy. *Geothermics*, **39**, 292 - 313, (2010)

Yeh, A., Croucher, A. E. and O'Sullivan, M. J.: Recent Developments in the Autough2 Simulator. *TOUGH Symposium 2012*, Ed. Lawrence Berkeley National Laboratory, Berkeley, California. (2012).

Zyvoloski, G. A.: *Fehm: A Control Volume Finite Element Code for Simulating Subsurface Multi-Phase Multi-Fluid Heat and Mass Transfer*. Los Alamos National Laboratory. Report LAUR-07-3359. (2007)